(2)

TECHNICAL REPORT RD-RE-89-5

AD-A207 994

MAXIMUM-LIKELIHOOD PARAMETER ESTIMATION OF A GENERALIZED GUMBEL DISTRIBUTION

Charles E. Hall, Jr.
Siegfried H. Lehnigk
Guttalu R. Viswanath
Research Directorate
Research, Development, and
Engineering Center

MARCH 1989

U.S. ARMY MISSILE COMMAND

Redstone Arsenal, Alabama 35898-5000

Cleared for public release; distribution is unlimited.

TECHNICAL REPORT RD-RE-89-5

MAXIMUM-LIKELIHOOD PARAMETER ESTIMATION OF A GENERALIZED GUMBEL DISTRIBUTION

Charles E. Hall, Jr.
Siegfried H. Lehnigk
and
Guttalu R. Viswanath*

Research Directorate
U.S. Army Missile Command
Redstone Arsenal, AL 35898-5248

MARCH 1989

Cleared for public release; distribution is unlimited.

*Work performed for the U.S. Army under U.S. Army Summer Faculty Research and Engineering Program. Permanent address: Department of Mathematics and Computer Science, South Carolina State College, Orangeburg, SC 29117

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No 0704 0188 Exp Date Jun 30 1986

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
2b DECLASSIFICATION/DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S) TR-RD-RE-89-5	5 MONITORING ORGANIZATION REPORT NUMBER(S)
6a NAME OF PERFORMING ORGANIZATION Research Directorate RD&E Center 6b OFFICE SYMBOL (If applicable) AMSMI-RD-RE-OP	7a NAME OF MONITORING ORGANIZATION
6c ADDRESS (City, State, and ZIP Code) U.S. Army Missile Command Redstone Arsenal, AL 35898-5248	7b ADDRESS (City, State, and ZIP Code)
8a NAME OF FUNDING/SPONSORING ORGANIZATION 8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO / PROJECT NO / TASK NO / WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification)

Maximum-Likelihood Parameter Estimation of a Generalized Gumbel Distribution

12 PERSONAL AUTHOR(S)

C. E. Hall, Jr., S. H. Lehnigk, and G. R. Viswanath

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Final	FROM May 88 TO Feb 89	1989/03/24	55

16 SUPPLEMENTARY NOTATION

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Extreme Value Distribution, Generalization, Density Estimation, Maximum-Likelihood Estimation

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

A microcomputer-based algorithm for estimation of the three parameters of a generalized Gumbel (extreme value type I) distribution class is presented. The parameters are shift, scale, and shape. The classical Gumbel distribution results if the shape parameter is equal to unity. Three-parameter as well as two-parameter (shape equal to unity) estimation can be performed for given histogram data. Parameter estimation is accomplished by means of the maximum-likelihood principle. The derivative equations which result from the associated logarithmic likelihood function are used. A more comprehensive presentation of generalized Gumbel distribution estimation which also allows treatment of population data and which includes moment estimates and maximum-likelihood estimates by direct optimization of the logarithmic likelihood function will be presented elsewhere.

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a NAME OF RESPONSIBLE INDIVIDUAL S. H. Lehnigk	22b TELEPHONE (Include Area Code) (205) 876-3526 22c OFFICE SYMBOL AMSMI-RD-RE-OP

DD FORM 1473, 84 MAR

83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

SUMMARY

A microcomputer-based algorithm for estimation of the three parameters of a generalized Gumbel (extreme value type I) distribution class is presented. The parameters are shift, scale, and shape. The classical Gumbel distribution results if the shape parameter is equal to unity. Three-parameter as well as two-parameter (shape equal to unity) estimation can be performed for given histogram data.

Parameter estimation is accomplished by means of the maximum-likelihood principle. The derivative equations which result from the associated logarithmic likelihood function are used. A more comprehensive presentation of generalized Gumbel distribution estimation which also allows treatment of population data and which includes moment estimates and maximum-likelihood estimates by direct optimization of the logarithmic likelihood function will be presented elsewhere.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF TABLES (Concluded)

I. INTRODUCTION

The Gumbel distribution (extreme value type I for maximum values) [5: Chap. 3.5], [6: Chap. 21.4] is defined by its probability density function (PDF)

$$f(x;\lambda,b) = b^{-1} \exp - (e^{-\xi} + \xi) \ , \ \xi = (x-\lambda)b^{-1} \ , x \in R \ , \tag{1.1}$$

or by its cumulative distribution function (CDF)

$$F(x;\lambda,b) = \exp - e^{-\xi} \ , \tag{1.2}$$

with parameters shift $\lambda \in R$, and scale $b > 0$. It is of considerable importance in various statistical data interpretation problems. Its CDF (1.2) forms the basis of a simple, highly popular, but not very reliable, linear regression procedure for the estimation of the parameters λ and b. With $u = \log(-\log F)$ it leads to the linear function

$$u = -b^{-1}x + \lambda b^{-1}.$$

Therefore, if a set of data $\{x_\nu, F_\nu\}$ is Gumbel distributed, the points $(x_\nu, u_\nu = \log(-\log F_\nu))$ are located on a straight line in the (x,u)-plane. In other words, if for a given data set $\{x_\nu, F_\nu\}$, the points (x_ν, u_ν) seem to be located on a straight line, the unknown parameters of the Gumbel PDF (1.1) can be determined by means of a least-squares algorithm [5: Chap. 8.1].

A more reliable parameter estimation approach can be based on the maximum-likelihood (ML) principle [14: Chap. 12.5], [8: Chap. 5.4] which leads to a simple algorithm relative to the Gumbel PDF (1.1). It is suggested, however, to apply the ML principle to a more general PDF class, namely

$$f(x;\lambda,\beta,b) = \frac{\exp(\beta \log \beta)}{b\,\Gamma(\beta)} \exp - \beta\left(e^{-\xi} + \xi\right) \ , \ \xi = (x-\lambda)b^{-1} \ , \ x \in R \ . \tag{*}$$

It depends on the parameters shift $\lambda \in R$, shape $\beta > 0$, and scale $b > 0$. Introduction of the shape parameter β makes the PDF class (*) much more flexible than the original Gumbel class (1.1) which is contained in (*) as a special case for shape $\beta = 1$.

The CDF of (*) is given by

$$F(x;\lambda,\beta,b) = \int_{-\infty}^{x} f(t;\lambda,\beta,b)\,dt = \frac{1}{\Gamma(\beta)} \Gamma\left(\beta, \beta \exp - \xi\right) \ ,$$

where $\Gamma(\alpha,z)$ is the incomplete Gamma function with lower integration limit z [3: 8.350.2]. Of course, this CDF leads to the linear relationship explained above only if $\beta = 1$, i.e., in general, a simple geometric parameter estimation technique is no longer

available. It may be of interest to note, however, that it is possible to show that log $(1 - \Gamma(b\xi - \lambda))$ is asymptotically linear in ξ as $\xi \uparrow \infty$. This can be seen by expressing the incomplete Gamma function in terms of the degenerate hypergeometric function [3; 8.351.2].

It is the objective of this report to show that parameter estimation for the three-parameter generalized Gumbel class (*) via the ML principle can be accomplished by means of a reliable algorithm which is more efficient than those currently available for the classical Gumbel class (1.1).

Questions of hypothesis justification and goodness of fit are outside the framework of this report and, thus, are not addressed here.

II. DIFFUSION CHARACTER OF THE GENERALIZED GUMBEL DISTRIBUTION

It is well known [9], [10], that the function

$$g(x,t) = \frac{\sigma}{c\,\Gamma\left((1-p)\sigma^{-1}\right)}\, \xi^{-p} \exp - \xi^{\sigma}\ ,\ \xi = xc^{-1}\ , \tag{II.1}$$

$x > 0$, $t > 0$, $\sigma > 0$, $p < 1$, with

$$c = c(t) = \begin{cases} \left[\alpha\sigma^2 t\right]^{1/\sigma}\ , & \tau = 0\ , \\ \left[\alpha\sigma\tau^{-1}\left(1 - \exp - \sigma\tau t\right)\right]^{1/\sigma}\ , & \tau \neq 0\ , \end{cases} \tag{II.2}$$

is the delta function initial condition (at $x = 0$, $t = 0$) solution of the autonomous Fokker-Planck equation

$$\left.\begin{aligned} &\left[A(x)\, z(x,t)\right]_{xx} - \left[D(x)\, z(x,t)\right]_{x} - z_t(x,t) = 0\ ,\ x > 0, t > 0\ , \\ &A(x) = \alpha x^{2-\sigma}\ ,\ \alpha > 0 \qquad \text{(diffusion coefficient)}, \\ &D(x) = \alpha(2-\sigma-p)x^{1-\sigma} - \tau x \qquad \text{(drift coefficient)}\ . \end{aligned}\right\} \tag{II.3}$$

With $t = t_0 > 0$ in (II.2) considered as a parameter, the function (II.1) becomes the hypergamma PDF with parameters $c = c(t_0) > 0$, $p < 1$, $\sigma > 0$ [10], [11].

The transformation $x = \exp - \gamma^{-1} y$ [5; (5.6)] generates from (II.1) the function

$$f(y,t) = g(\exp - \gamma^{-1} y, t)\ |dx/dy|$$

which, after y has been replaced by x, is of the form

$$f(x,t) = \frac{1}{b\,\Gamma(\beta)\, c^{\sigma\beta}}\, \exp - \left(c^{-\sigma} e^{-\xi} + \beta\xi\right)\ ,\ \xi = xb^{-1}\ , \tag{II.4}$$

$b = \sigma^{-1}\gamma$, $\beta = (1-p)\sigma^{-1}$, and $c = c(t)$ given in (II.2). For any fixed $t = t_0 > 0$ and with $c(t_0) = \beta^{-1/\sigma}$, this function becomes identical with (1.4) (for $\lambda = 0$).

The transformation $x = \exp - \gamma^{-1}y$ applied to the differential equation (II.3) leads (after y has again been replaced by x) to the autonomous Fokker-Planck equation

$$\left.\begin{aligned} \left[A^*(x)\,w(x,t)\right]_{xx} - \left[D^*(x)\,w(x,t)\right]_x - w_t(x,t) &= 0\ , \\ A^*(x) &= \alpha\sigma^2 b^2 \exp b^{-1}x\ , \\ D^*(x) &= \alpha\sigma^2 b\,(1-\beta)\exp b^{-1}x + \sigma\tau b\ , \end{aligned}\right\} \qquad \text{(II.5)}$$

of which the function (II.4) is a solution. By means of different parameter designations the function (II.4) can also be transformed so that it becomes a nonlinear similarity solution of (II.5) [12: Secs. 2,3].

The particular case $1 - p = \sigma$ reduces the PDF (II.1) to the Weibull PDF [11], and (II.4) becomes the Gumbel PDF with $\beta = 1$.

III. DATA SETS

It will be assumed that the given statistical data are of the general form of a set of ordered pairs $\{x_\nu, f_{a\nu})\}$ $(\nu=1, \ldots M)$ with $x_1 < x_2 < \ldots < x_M$, and $M \le N = \Sigma^M{}_{\nu=1} f_{a\nu}$, $f_{a\nu}$ being the absolute (integer valued) frequency of the observation x_ν. In standard statistical terminology data given in this particular form are called histogram data.

It will furthermore be assumed that, for histogram estimation, the x_ν's are the midpoints of class intervals $[a + (\nu-1)\Delta a\ ,\ a + \nu\Delta a)$, $a \in R$, $\Delta a > 0$, i.e., $x_\nu = a + (\nu-1/2)\Delta a$, and that $f_{a\nu} \ge 0$, $f_{a1} \ge 1$, $f_{aM} \ge 1$. Other estimation problems, such as population estimation and different approaches will be presented elsewhere.

IV. THE LOGARITHMIC LIKELIHOOD FUNCTION

For the generalized Gumbel PDF (*) with parameter vector $P = (\lambda, \beta, b)$ and with the abbreviation

$$\rho_\nu(\lambda) = x_\nu - \lambda$$

the likelihood function takes the particular form

$$L(P) = \beta^{N\beta} b^{-N} \Gamma^{-N}(\beta) \exp - \beta \left[\sum_{\nu=1}^{M} f_{a\nu} \exp - b^{-1} \rho_\nu + \sum_{\nu=1}^{M} f_{a\nu} b^{-1} \rho_\nu \right] .$$

The function

$$\Phi(P) = N^{-1} \log L(P)$$

$$= \beta \log \beta - \log b - \log \Gamma(\beta) - \beta \sum_{\nu=1}^{M} f_\nu \left(b^{-1} \rho_\nu + \exp - b^{-1} \rho_\nu \right) , \qquad \text{(IV.1)}$$

where $f_\nu = N^{-1} f_{a\nu}$ denotes the relative frequencies of the x_ν's with $\Sigma^M_{\nu=1} f_\nu = 1$, is the *logarithmic likelihood function (LLF) of the generalized Gumbel distribution class* (*).

The ML principle asserts that the optimal parameter values (if they exist) relative to the given data are the coordinates of the point $\hat{P} = (\hat{\lambda}, \hat{\beta}, \hat{b})$ in the open parameter space $\mathcal{P}: \{\lambda \in R, \beta > 0, b > 0\}$ at which the LLF $\phi(P)$ takes it global maximum.

V. MAXIMUM-LIKELIHOOD ESTIMATION

In this report ML estimation of the parameters of the PDF class (*) will be done by means of the derivative equations which result from equating to zero the first partial derivatives of the LLF $\Phi(P)$ given in (IV.1). They are of the form

$$\frac{\partial\Phi}{\partial\lambda} = \beta b^{-1}\left[1-\sum_{\nu=1}^{M} f_\nu \exp - b^{-1}\rho_\nu\right] = 0 \;,$$

$$\frac{\partial\Phi}{\partial\beta} = \log\beta + 1 - \psi(\beta) - \sum_{\nu=1}^{M} f_\nu\left(b^{-1}\rho_\nu + \exp - b^{-1}\rho_\nu\right) = 0 \;,$$

$$\frac{\partial\Phi}{\partial b} = -b^{-1} + \beta b^{-2}\sum_{\nu=1}^{M} f_\nu\rho_\nu\left(1-\exp - b^{-1}\rho_\nu\right) = 0 \;,$$

where $\psi(x) = d \log\Gamma(x)/dx$ is the psi function [3; 8.362]. Their appearance can be simplified by introduction of the functions

$$A(\lambda,b) = \sum_{\nu=1}^{M} f_\nu \exp - b^{-1}\rho_\nu > 0 \;,$$

$$B(\lambda,b) = \sum_{\nu=1}^{M} f_\nu x_\nu \exp - b^{-1}\rho_\nu > 0.$$

Then, with the mean

$$\bar{x} = \sum_{\nu=1}^{M} f_\nu x_\nu = \text{const.} \;,\quad 0 < \bar{x} < x_M \;,$$

they take the form

$$1 - A = 0 \;,$$

$$\log\beta + 1 - \psi(\beta) - b^{-1}(\bar{x}-\lambda) - A = 0 \;,$$

$$-b + \beta(\bar{x} - \lambda - B + \lambda A) = 0 \;. \qquad \text{(V.1)}$$

By means of the first one, A can be eliminated from the other two so that only the equations

$$r(\beta) - b^{-1}(\bar{x} - \lambda) = 0 \; ,$$

$$-b + \beta(\bar{x} - B) = 0 \; , \qquad \text{(V.2)}$$

remain. The function

$$r(\beta) = \log \beta - \psi(\beta)$$

which appears here is positive for $\beta > 0$ and is strictly convex and decreasing, $r(\beta) \uparrow \infty$ as $\beta \downarrow 0$, $r(\beta) \downarrow 0$ as $\beta \uparrow \infty$, as can be verified by means of the Laplace integral representation for $r(\beta)$ [3: 8.361.8]. The first one of these equations allows one to express b in terms of β and λ,

$$b = b(\lambda,\beta) = \left(\bar{x} - \lambda\right)/r(\beta) > 0 \; . \qquad \text{(V.3)}$$

Thus, b can be eliminated from equations (V.1) and (V.2). Setting

$$A(\lambda,b) = A(\lambda,b(\lambda,\beta)) = Q(\lambda,\beta) = \sum_{\nu=1}^{M} f_\nu T_\nu(\lambda,\beta) > 0 \; , \qquad \text{(V.4)}$$

$$B(\lambda,b) = B(\lambda,b(\lambda,\beta)) = R(\lambda,\beta) = \sum_{\nu=1}^{M} f_\nu T_\nu(\lambda,\beta) > 0 \; , \qquad \text{(V.5)}$$

with

$$T_\nu(\lambda,\beta) = \exp - \left[\rho_\nu (\bar{x} - \lambda)^{-1} r(\beta) \right] \; (\nu = 1, \ldots, M)$$

one obtains

$$g(\lambda,\beta) = \left[\bar{x} - R(\lambda,\beta)\right] \beta \, r(\beta) - (\bar{x} - \lambda) = 0 \qquad \text{(V.6)}$$

$$h(\lambda,\beta) = 1 - Q(\lambda,\beta) = 0 \; . \qquad \text{(V.7)}$$

If $(\hat{\lambda}, \hat{\beta})$ is a solution of this system of equations, then $\hat{b}$ follows from (V.3) for $\lambda = \hat{\lambda}$, $\beta = \hat{\beta}$. The objective, therefore, is to develop an efficient numerical algorithm for the solution of equations (V.6) and (V.7). Properties of the functions $g(\lambda,\beta)$ and $h(\lambda,\beta)$ which are essential for this purpose will be investigated in Section VI.

Two observations relative to the shift parameter λ are of importance. Relation (V.3) requires that $\lambda < \bar{x}$ for the scale parameter to be positive. Furthermore, if $\lambda \leq x_1 = \min\{x_\nu\}$, then $(x_\nu - \lambda)(\bar{x} - \lambda)^{-1} > 0$ for $\nu \geq 2$ so that $Q(\lambda,\beta) < 1$ which contradicts (V.7). Consequently, in (V.6) and (V.7), λ is restricted to the interval $x_1 < \lambda < \bar{x}$). It should also be observed that the positive function $\beta r(\beta)$ which appears in (V.6) is less than unity for $\beta > 0$. This follows from the integral representation for $r(\beta)$ mentioned earlier.

The special case that the shape parameter β is assumed to be unity will be discussed next. In this situation, the LLF (IV.1) reduces to

$$\Phi(\lambda,b) = -\log b - \sum_{\nu=1}^{M} f_\nu \left(b^{-1} \rho_\nu + \exp - b^{-1} \rho_\nu \right).$$

The derivative equations now lead to

$$1 - A = 0 ,$$
$$-b - B + \bar{x} = 0 ,$$

with A and B, functions of x and b, defined as before. The first equation can be solved for λ,

$$\lambda = -b \log \sum_{\nu=1}^{M} f_\nu \exp - b^{-1} x_\nu . \qquad \text{(V.8)}$$

and λ can be eliminated from B so that

$$B = \left[\sum_{\nu=1}^{M} f_\nu \exp - b^{-1} x_\nu \right]^{-1} \sum_{\nu=1}^{M} f_\nu x_\nu \exp - b^{-1} x_\nu .$$

Therefore, the second equation becomes an equation in just one unknown,

$$g(b) = \bar{x} - b - \left[\sum_{\nu=1}^{M} f_\nu \exp - b^{-1} x_\nu \right]^{-1} \sum_{\nu=1}^{M} f_\nu x_\nu \exp - b^{-1} x_\nu = 0 . \qquad \text{(V.9)}$$

It is easily seen that $g'(b)$ is negative for $b > 0$ and that $g(b) \uparrow \bar{x} - x_1 > 0$ as $b \downarrow 0$, and $g(b) \downarrow -\infty$ as $b \uparrow \infty$. It should also be observed that (V.9) implies that $\hat{b} < \bar{x}$ if $g(\hat{b}) = 0$.

By the property of the function $g(b)$ just discussed, equation (V.9) has exactly one positive root $\hat{b}$. It immediately gives the corresponding value $\hat{\lambda}$ for the shift parameter from (V.8) for $b = \hat{b}$. ML estimation for the two-parameter Gumbel distribution is, therefore, easily accomplished by the solution of the single equation (V.9).

VI. PROPERTIES OF $g(\lambda,\beta)$ AND $h(\lambda,\beta)$

This section contains a description of properties of the functions $g(\lambda,\beta)$ and $h(\lambda,\beta)$ which are essential for the design of an efficient solution algorithm for the system of equations (V.6) and (V.7). The function $h(\lambda,\beta)$ will be considered first.

By means of the substitution

$$y = \exp - \left(\bar{x} - \lambda\right)^{-1} r(\beta) \ , \qquad \text{(VI.1)}$$

which takes the functions $T_\nu(\lambda,\beta)$ appearing in (V.4) and (V.5) into

$$T_\nu(\lambda,\beta) = y^{x_\nu - \lambda} \ ,$$

equation (V.7) changes into

$$h(\lambda,\beta) = 1 - y^{-\lambda} \sum_{\nu=1}^{M} f_\nu y^{x_\nu} = 0 \ . \qquad \text{(VI.2)}$$

It is equivalent to the equation

$$f(y) = y^{\lambda - x_1} - \sum_{\nu=1}^{M} f_\nu y^{x_\nu - x_1} = u(y) - v(y) = 0 \ .$$

For fixed λ in the interval $(x_1, \bar{x})$, $y = y(\beta)$ becomes a one-to-one mapping of $(0,\infty)$ into $(0,1)$. Clearly, $f(y) \to 0$ as $y \uparrow 1$ (i.e., as $\beta \uparrow \infty$). Furthermore,

$$u'(1) = \lambda - x_1 > 0 \ ,$$

$$v'(1) = \sum_{\nu=2}^{M} f_\nu \left(x_\nu - x_1\right) = \bar{x} - x_1 > 0 \ .$$

Since $\bar{x} - x_1 > \lambda - x_1 > 0$, it follows that $v'(1) > u'(1) > 0$. This implies that $0 < v(y) < u(y)$ for $y < 1$, y sufficiently close to 1. Consequently $f(y) > 0$ for $y < 1$, y sufficiently close to 1, and, hence, $h(\lambda,\beta) > 0$ for β sufficiently large. Furthermore, $h(\lambda,\beta) \downarrow 0$ as $\beta \uparrow \infty$.

Next, $u(y) \downarrow 0$, $v(y) \downarrow f_1$, as $y \downarrow 0$. Therefore, $f(y) \to -f_1$ as $y \downarrow 0$, and hence, $h(\lambda,\beta) \downarrow -\infty$ as $\beta \downarrow 0$. Consequently, for every fixed $\lambda \in (x_1, \bar{x})$, the function $h(\lambda,\beta)$ has an odd number of positive zeros.

It can now be shown that, for every fixed $\lambda \in (x_1, \bar{x})$, equation (V.7) has exactly one positive simple root $\hat{\beta}$. The derivative of $h(\lambda,\beta)$ with respect to β is of the form

$$\frac{\partial h}{\partial \beta} = -\frac{r'}{\bar{x}-\lambda}\left(\lambda Q - R\right) = -\frac{r'}{\bar{x}-\lambda}\, q(\beta)\,. \tag{VI.3}$$

R and Q defined in (V.4) and (V.5), respectively. The factor $-(\bar{x}-\lambda)^{-1} r'$ is positive for every $\beta > 0$. Suppose the second factor

$$q(\beta) = -\sum_{\nu=1}^{M} f_\nu T_\nu \left(x_\nu - \lambda\right)$$

has two distinct positive zeros, $0 < \beta_1 < \beta_2$. By Rolle's Theorem, $q'(\beta)$ must have a zero in the interval (β_1, β_2). But

$$q'(\beta) = -r' \sum_{\nu=1}^{M} f_\nu T_\nu \frac{\left(x_\nu - \lambda\right)^2}{\bar{x}-\lambda} > 0$$

for every $\beta > 0$. Therefore, $q(\beta)$ has at most one positive zero. By properties of h discussed earlier, $\partial h/\partial\beta$ has at least one positive zero. Therefore, the function $q(\beta)$, which is equivalent to $\partial h/\partial\beta$, has exactly one positive zero, say, β_0. Since h has an odd number of positive zeros and since $h(\lambda,\beta) < 0$ for small values of $\beta > 0$ and $h(\lambda,\beta) > 0$ for large values of β, it follows that $h(\lambda,\beta)$ has exactly one positive zero, say, $\beta_h(\lambda)$, such that $0 < \beta_h(\lambda) < \beta_0$. The zero β_h is simple since $q(\beta_h) \neq 0$. For, if $q(\beta_h) = 0$, $q(\beta)$ would have two distinct zeros.

It is now possible to show that the positive function $\beta_h(\lambda)$, the unique zero of $h(\lambda,\beta)$ for given $\lambda \in (x_1, \bar{x})$, implicitly defined by equation (V.7), is strictly monotonically increasing in the interval $(x_1, \bar{x})$. Let $\hat{\lambda}$ and $\hat{\beta} = \beta_h(\hat{\lambda})$ be such that $h(\hat{\lambda}, \hat{\beta}) = 0$. By the implicit function theorem there exists a function $\beta_h(\lambda)$ such that $h(\lambda,\beta_h(\lambda)) = 0$ on some interval (λ_1, λ_2) which contains $\hat{\lambda}$. Since $q(\beta_h(\lambda))$ ($q(\beta)$ having been defined in (VI.3)) is different from zero, in fact

$$q(\beta_h(\lambda)) = \lambda Q(\lambda, \beta_h(\lambda)) - R(\lambda, \beta_h(\lambda)) = \lambda - R(\lambda, \beta_h(\lambda)) > 0 \tag{VI.4}$$

for every $\lambda \in (x_1, \bar{x})$, the function $\beta_h(\lambda)$ can be continued from (λ_1, λ_2) as a differentiable function to the entire interval $(x_1, \bar{x})$.

Its derivative is defined by the identity

$$\frac{d}{d\lambda} h\left(\lambda, \beta_h(\lambda)\right) = \frac{r\left(\beta_h(\lambda)\right)}{\left(\bar{x}-\lambda\right)^2}\left[\bar{x} - R\left(\lambda, \beta_h(\lambda)\right)\right] + \frac{r'\left(\beta_h(\lambda)\right)}{\bar{x}-\lambda}\left[\lambda - R\left(\lambda, \beta_h(\lambda)\right)\right]\frac{d\beta_h(\lambda)}{d\lambda} = 0. \tag{VI.5}$$

Since $\Sigma^M_{\nu=1} f_\nu T_\nu (\lambda, \beta_h(\lambda)) \equiv 1$, it follows from Tchebychef's inequality that $\bar{x} - R(\lambda, \beta_h(\lambda)) > 0$ for every $\lambda \in (x_1, \bar{x})$. Furthermore, by (VI.4), $\lambda - R(\lambda, \beta_h(\lambda)) > 0$. Therefore, (VI.5) implies that $d\beta_h(\lambda)/d\lambda > 0$ for $\lambda \in (x_1, \bar{x})$. That is to say, $\beta_h(\lambda)$ is strictly monotonically increasing in the interval $(x_1, \bar{x})$.

The following facts concerning the function $\beta_h(\lambda)$ should finally be observed. By (VI.5), its derivative is given by

$$\frac{d\beta_h(\lambda)}{d\lambda} = \frac{r\left(\beta_h(\lambda)\right)\left[\bar{x} - R\left(\lambda, \beta_h(\lambda)\right)\right]}{-r'\left(\beta_h(\lambda)\right)\left(\bar{x}-\lambda\right)\left[\lambda - R\left(\lambda, \beta_h(\lambda)\right)\right]} . \tag{VI.6}$$

The positive function $[\bar{x} - R(\lambda, \beta_h(\lambda))]\,[\lambda - R(\lambda, \beta_h(\lambda))]^{-1} \downarrow 1$ as $\lambda \uparrow \bar{x}$. The positive function $r(\beta_h(\lambda))\,[-r'(\beta_h(\lambda))]^{-1}$ can be expressed in the form

$$\frac{r(x)}{-r'(x)} = \frac{xr(x)}{x\psi'(x) - 1} . \tag{VI.7}$$

Since $\beta_h(\lambda)$ is positive and strictly monotonically increasing, the argument $x = \beta_h(\lambda)$ in (VI.7) cannot go to zero as $\lambda \uparrow \bar{x}$. Furthermore, for $x > 0$, the numerator $xr(x)$ in (VI.7) is bounded above by unity, and $xr(x) \downarrow 1/2$ as $x \downarrow \infty$. This limit relation can be established by means of the Laplace integral representation of $r(x)$ mentioned before and by application of Theorem 14.1.3 in [2]. The denominator function in (VI.7) is positive for $x > 0$, and $x\psi'(x) \downarrow 0$ as $x \uparrow \infty$. This limit relation can be verified by use of an integral representation of $\psi'(x)$ [15; Chap. 12.32]. Consequently, as (VI.6) shows, $\beta'_h(\lambda) \uparrow \infty$ as $\lambda \uparrow \bar{x}$, i.e., $\beta_h(\lambda)$ has a singularity at $\lambda = \bar{x}$.

Now suppose $\beta_h(\lambda)$ would approach a finite limit as $\lambda \uparrow \bar{x}$. Then, as (VI.1) shows, $y \downarrow 0$ as $\lambda \uparrow \bar{x}$, i.e., the second term in (VI.2) will not remain finite as $\lambda \uparrow \bar{x}$. This establishes a contradiction to the identity (VI.2) with $\beta = \beta_h(\lambda)$ and λ sufficiently close to $\bar{x}$. Therefore, $\beta_h(\lambda) \uparrow \infty$ as $\lambda \uparrow \bar{x}$. For this reason, numerical solution of equation (V.7) may fail for λ close to $\bar{x}$.

Equation (V.6) will now be considered. The transformation (VI.1) changes it into

$$g(\lambda,\beta) = -\left[\bar{x} - y^{-\lambda} \sum_{\nu=1}^{M} f_\nu x_\nu y^{x_\nu}\right] \beta(y) \log y - 1 = 0 .$$

An equivalent equation is

$$\left[\bar{x}\, y^{\lambda-x_1} - \sum_{\nu=1}^{M} f_\nu x_\nu y^{x_\nu - x_1}\right] \beta(y) \log y + y^{\lambda-x_1} = 0 \qquad \text{(VI.8)}$$

in which λ is considered fixed in the interval $(x, \bar{x})$. The function $\beta(y)$ cannot be *explicitly expressed in terms of* y. *However, it is only necessary to know that*

$$\beta(y) \log y = -\left(\bar{x}-\lambda\right)^{-1} \beta r(\beta) \rightarrow \begin{cases} -\left(\bar{x}-\lambda\right)^{-1} & \text{as } \beta \downarrow 0, \text{ i.e., as } y \downarrow 0, \\ -\left[2\left(\bar{x}-\lambda\right)\right]^{-1} & \text{as } \beta \uparrow \infty, \text{ i.e., as } y \uparrow 1. \end{cases}$$

The second limit relation has essentially been established already in connection with (VI.7). The first one follows from the fact that $xr(x) \uparrow 1$ as $x \downarrow 0$ [3; 8.362.2].

It can now be seen that the left-hand side of (VI.8) approaches positive values as $y \downarrow 0$ and as $y \uparrow 1$. This implies that, for fixed $\lambda \in (x_1, \bar{x})$, the equation $g(\lambda,\beta) = 0$ has an even number of positive roots. There may exist values of $\lambda \in (x_1, \bar{x})$ for which $g(\lambda,\beta) = 0$ has no positive roots at all. As a matter of fact, a more careful investigation of equation (VI.8) reveals that there may exist numbers λ_1 and λ_2, $x_1 < \lambda_1 < \lambda_2 < \bar{x}$, such that $g(\lambda,\beta) = 0$ has no real roots for $\lambda \in (\lambda_1, \lambda_2)$.

VII. THE PARAMETER ESTIMATION ALGORITHM

Estimation of the parameters of the generalized Gumbel distribution class (*), Section I, for a given set of histogram data $\{(x_\nu, f_\nu\}$ $(\nu=1, \ldots, M)$, $x_\nu = a + (\nu - 1/2)\Delta a$, is based on the solution of the derivative equations (Section V) of the associated LLF (IV.1). The algorithm presented in this report covers the three-parameter unknown case as well as the two-parameter unknown case in which the shape parameter β is assumed to be equal to unity.

In the three-parameter case the equations $g(\lambda,\beta) = 0$ (V.6), and $h(\lambda,\beta) = 0$ (V.7) must be solved. Their solution $(\hat{\lambda}, \hat{\beta})$ determines the scale parameter $\hat{b}$ by (V.3). In the two-parameter case the single equation $g(b) = 0$ (V.9) must be solved. Its solution $\hat{b}$ determines the shift parameter $\hat{\lambda}$ by (V.8).

1. The Three-Parameter Case

Suppose that $(\hat{\lambda}, \hat{\beta})$ is a solution of the system of equations (V.6) and (V.7) with $\hat{\lambda} \in (x_1, \bar{x})$ and $\hat{\beta} = \beta_h(\hat{\lambda}) > 0$, and that the graphs of the functions implicitly defined by $g = 0$ and $h = 0$ have a proper intersection at $(\hat{\lambda}, \hat{\beta})$. Then there are numbers λ_L and λ_R , $x_1 < \lambda_L < \hat{\lambda} < L_R < \bar{x}$, and numbers $\beta_L = \beta_h(\lambda_L)$ and $\beta_R = \beta_h(\lambda_R)$, $0 < \beta_L < \beta_R$, such that $h(\lambda_L, \beta_L) = 0$, $h(\lambda_R, \beta_R) = 0$ and $g(\lambda_L, \beta_L)\, g(\lambda_R, \beta_R) < 0$. Since $\beta_h(\lambda)$ is strictly monotonically increasing (Section VI), the solution point $(\hat{\lambda}, \hat{\beta})$ of the equations (V.6) and (V.7) is boxed in the interior of the rectangle in the first quadrant of the (λ,β) plane defined by the points (λ_L, β_L), (λ_L, β_R) , (λ_R, β_L), (λ_R, β_R).

Boxing of the solution point $(\hat{\lambda}, \hat{\beta})$ requires the solution of the equation $h(\lambda,\beta) = 0$ only and evaluation of the sign of the function $g(\lambda,\beta)$.

The root $\beta_L(\lambda)$ of $h(\lambda,\beta) = 0$ is first bracketed by sign change detection along the search sequence $\beta_\nu = 10^{-1}\,(1.618)^\nu$ $(\nu=0, 1, 2, \ldots, 19)$. Brent's method [13; Chap. 7.3] is then used to calculate the root. In the numerical algorithm it is actually not the equation $h = 0$ which is solved but the equivalent equation $f = 0$ given in Sec. VI. The range of f is bounded for $\beta \in (0,\infty)$ whereas the range of h is not bounded below near $\beta = 0$. The use of f instead of h eliminates possible overflow problems for small values of β . The parameter estimation algorithm is aborted if no root $\beta_h(\lambda)$ is found in the interval $(10^{-1}, \beta_{19})$. Shape parameter values outside this interval would lead to nearly pathological densities.

To obtain λ values for boxing the solution of equations (V.6) and (V.7), the β roots of $h(\lambda,\beta) = 0$ are calculated along a sequence $\{\lambda^{(0)}, \lambda^{(0)}{}_R, \lambda^{(0)}{}_L, \lambda^{(1)}{}_R, \lambda^{(1)}{}_L, \ldots\}$ of λ values with $\lambda^{(0)} = \bar{x} - \Delta a/2$ and

$$\lambda_R^{(\nu)} = \lambda^{(0)} + 2^{\nu-1}\,\Delta a\left(2^{\nu} + 99\right)^{-1} \ .$$

$$\lambda_L^{(\nu)} = \lambda^{(0)} - 2^{\nu}\left(\lambda^{(0)} - x_1\right)\left(2^{\nu} + 99\right)^{-1} \quad (\nu = 0, 1, 2, \ldots, 30) \ .$$

The first of these subsequences converges up to $\bar{x}$, the second down to x_1 as $\nu \uparrow \infty$. Let $\beta^{(0)}$, $\beta^{(\nu)}{}_R$, $\beta^{(\nu)}{}_L$, be the corresponding roots of $h = 0$.

The sign of $g(\lambda,\beta)$ is evaluated at the points $(\lambda^{(0)}, \beta^{(0)})$, $(\lambda^{(\nu)}{}_R, \beta^{(\nu)}{}_R)$, and $(\lambda^{(\nu)}{}_L, \beta^{(\nu)}{}_L)$. The process is terminated if at two successive points, either in the $\{\lambda^{(\nu)}{}_R\}$ or in the $\{\lambda^{(\nu)}{}_L\}$ subsequence, the product of the g functions is negative, i.e., if boxing of $(\hat{\lambda}, \hat{\beta})$ has been achieved.

The boxing rectangle is subsequently refined by bisection of the λ interval until the accuracy requirement $(\Delta\lambda < 10^{-7}) \cap (\Delta\beta < 10^{-7})$ is satisfied. Monotonicity of the function $\beta_h(\lambda)$ is not exploited in the actual numerical algorithm because of the interplay between the one-dimensional Brent's method and the two-dimensional boxing process.

2. The Two-Parameter Case

The root $\hat{b}$ of equation (V.9), $g(b) = 0$, is first bracketed by evaluation of $g(b_\nu)$ along the search sequence $b_\nu = 10^{-1}\,(1.618)^{\nu}$ $(\nu=0,1,2, \ldots, 19)$. The reason for the restriction of b to this interval is analogous to that given earlier for β.

Remark. For computational convenience the transformation $x_\nu \to x_\nu - x_1$ is applied before the start of the actual computations. It affects only the shift parameter. The algorithm calculates the relative shift value $\hat{\lambda}_0$ and returns the true shift value $\hat{\lambda} = \hat{\lambda}_0 + x_1$.

VIII. EXAMPLES

To demonstrate the algorithm a number of examples are presented. Accompanying tables are given in Section IX.

Example 1

N = 87 observations of annual 24-hour maximum rainfalls (in points) at Sidney, Australia, over the period 1859-1945 are given in Table 1.1 [7]. Grouping into histogram absolute frequency data has been performed on the data of Table 1.1 in five different ways, G_ν, as displayed in the second columns of Tables 1.3G_ν (ν=1, ... , 5). The first column shows the class interval numbers ν. The class interval data are given in Table 1.2.

Parameter estimation has been performed with three and with two (β=1) parameters. The results are shown in Tables 1.4G_ν (ν = 1, ... , 5).

The estimated parameter values are used to calculate the expected absolute frequencies from the PDF (*). For the three-parameter estimates they are given in the third columns of Tables 1.3G_ν, and for the two-parameter estimates in the fifth columns. Next to the calculated frequencies in Tables 1.3G_ν are shown the chi-square values. As mentioned in the Introduction, no significance will be attached to them within the framework of this report.

Example 2

A total of N = 89 observations of frequencies of annual maxima of rainfall (in inch) in 24 hours at Camden Square, London, over the period 1860-1948 [1; Chap. 8.7] have been grouped into M = 12 class intervals $[a + (\nu-1)\Delta a, a + \nu\Delta a)$ with a = 1/2, Δa = 1/4. The data are given in Table 2.1. The results of the three-parameter and two-parameter estimates are shown in Table 2.2. The calculated expected frequencies together with the corresponding chi-square values are displayed in Table 2.1.

Example 3

This example concerns the distribution of the greatest ages of men [4; Sec. 1, Table I, 2nd ed.]. A total of N = 52 observations have been grouped into M = 9 class intervals $[a + (\nu-1)\Delta a, a + \nu\Delta a)$ with a = 95.5 and Δa = 1. Table 3.1 shows the given data. Calculated results are shown in Table 3.2 (parameters) and in Table 3.1 (frequencies and chi-square values).

Example 4

The last example deals with the ensemble distribution of the greatest ages of men and women [4; Sec. 1, Table I, 4th col.]. The total number of observations is N = 104 corresponding to M = 11 class intervals $[a + (\nu-1)\Delta a, a + \nu\Delta a)$, a = 95.5, Δa = 1. Table 4.1 shows the given data. The results are presented in Tables 4.2 and 4.1.

IX. TABLES

Table 1.1 Annual Maximum Twenty-Four Hour Rainfalls (in Points) at Sidney, Australia, 1859-1945.

370	752	662	190	375	395	395	301
565	618	445	403	324	280	890	342
189	281	489	753	569	204	425	646
433	645	485	468	283	275	636	566
295	434	330	310	236	487	273	605
301	423	441	637	157	477	364	363
367	571	177	475	708	441	652	337
325	342	316	653	488	414	418	374
250	188	484	459	489	239	391	302
280	322	380	335	263	216	339	375
230	333	154	1105	336	192	420	

Table1.2 Class Intervals and Number of Classes for Groupings G_ν ($\nu=1, \ldots, 5$)

	G_1	G_2	G_3	G_4	G_5
a	150	125	150	100	125
Δa	100	100	50	75	75
M	10	10	20	14	14

Table 1.3G_1 Grouped Frequency Data for Grouping G_1

GUMBEL DISTRIBUTION CALCULATIONS
Data file is LNGDSET1.DAT

		3 Para Estimate		2 Para Estimate	
v	f(abs)	f(cal)	X**2	f(cal)	X**2
1	11	10.41	.03	9.56	.22
2	23	25.08	.17	22.20	.03
3	23	20.91	.21	22.35	.02
4	10	13.10	.73	15.15	1.75
5	10	7.60	.76	8.51	.26
6	4	4.32	.02	4.37	.03
7	4	2.44	1.00	2.15	1.58
8	1	1.38	.10	1.04	.00
9	0	.78	.78	.50	.50
10	1	.44	.72	.24	2.46
Totals	87	86.43		86.08	

Table 1.3G_2 Grouped Frequency Data for Grouping G_2

GUMBEL DISTRIBUTION CALCULATIONS
Data file is LNGDSET2.DAT

		3 Para Estimate		2 Para Estimate	
v	f(abs)	f(cal)	X**2	f(cal)	X**2
1	8	6.79	.22	6.79	.22
2	18	20.64	.34	19.70	.15
3	25	22.61	.25	22.74	.23
4	16	16.10	.00	16.78	.04
5	7	9.56	.68	9.97	.89
6	7	5.25	.58	5.34	.52
7	3	2.79	.02	2.71	.03
8	2	1.47	.19	1.35	.32
9	0	.77	.77	.66	.66
10	1	.40	.91	.32	1.43
Totals	87	86.38		86.35	

Table 1.3G_3 Grouped Frequency Data for Grouping G_3

GUMBEL DISTRIBUTION CALCULATIONS
Data file is LNGDSET3.DAT

		3 Para Estimate		2 Para Estimate	
v	f(abs)	f(cal)	X**2	f(cal)	X**2
1	6	3.13	2.63	3.20	2.46
2	5	7.28	.71	6.74	.45
3	8	10.80	.72	10.04	.42
4	15	12.15	.66	11.77	.89
5	12	11.56	.02	11.69	.01
6	11	9.90	.12	10.36	.04
7	10	7.95	.53	8.49	.27
8	0	6.14	6.14	6.59	6.59
9	5	4.63	.03	4.93	.00
10	5	3.44	.71	3.59	.55
11	3	2.53	.09	2.57	.07
12	1	1.85	.39	1.82	.37
13	3	1.35	2.02	1.28	2.32
14	1	.98	.00	.89	.01
15	1	.71	.11	.62	.23
16	0	.52	.52	.43	.43
17	0	.38	.38	.30	.30
18	0	.27	.27	.21	.21
19	0	.20	.20	.14	.14
20	1	.14	5.11	.10	8.34
Totals	87	85.93		85.74	

Table 1.3G_4 Grouped Frequency Data for Grouping G_4

GUMBEL DISTRIBUTION CALCULATIONS
Data file is LNGDSET4.DAT

		3 Para Estimate		2 Para Estimate	
v	f(abs)	f(cal)	X**2	f(cal)	X**2
1	2	1.84	.01	2.08	.00
2	9	8.91	.00	8.55	.02
3	15	16.12	.08	15.46	.01
4	20	17.39	.39	17.35	.40
5	13	14.32	.12	14.76	.21
6	8	10.23	.48	10.70	.68
7	7	6.78	.01	7.06	.00
8	6	4.33	.65	4.41	.57
9	3	2.70	.03	2.67	.04
10	2	1.67	.06	1.59	.10
11	1	1.03	.00	.94	.00
12	0	.63	.63	.55	.55
13	0	.39	.39	.32	.32
14	1	.24	2.47	.19	3.56
Totals	87	86.57		86.62	

Table 1.3G_5 Grouped Frequency Data for Grouping G_5

GUMBEL DISTRIBUTION CALCULATIONS
Data file is LNGDSET5.DAT

		3 Para Estimate		2 Para Estimate	
v	f(abs)	f(cal)	X**2	f(cal)	X**2
1	6	3.98	1.02	4.05	.93
2	8	12.24	1.47	11.76	1.20
3	20	17.51	.35	17.16	.47
4	17	16.70	.01	16.88	.00
5	16	12.82	.79	13.23	.58
6	4	8.78	2.60	9.09	2.85
7	6	5.66	.02	5.80	.01
8	4	3.53	.06	3.54	.06
9	3	2.16	.32	2.11	.38
10	1	1.31	.07	1.24	.05
11	1	.79	.05	.72	.11
12	0	.48	.48	.42	.42
13	0	.29	.29	.24	.24
14	1	.17	3.94	.14	5.33
Totals	87	86.43		86.38	

Table 1.4G_1 Three- and Two-Parameter Estimates for Grouping G_1

```
Input file is LNGDSET1.DAT
This is a  3 parameter fit
*****************************************
THE FINAL VALUES

Lambda =     .316619E+003
  b    =     .762429E+002
Beta   =     .436916E+000
*****************************************

Input file is LNGDSET1.DAT
This is a  2 parameter fit
*****************************************
THE FINAL VALUES

Lambda =     .348179E+003
  b    =     .134030E+003
Beta   =     .100000E+001
*****************************************
```

Table 1.4G_2 Three- and Two-Parameter Estimates for Grouping G_2

```
Input file is LNGDSET2.DAT
This is a  3 parameter fit
*****************************************
THE FINAL VALUES

Lambda =     .336618E+003
  b    =     .116041E+003
Beta   =     .759879E+000
*****************************************

Input file is LNGDSET2.DAT
This is a  2 parameter fit
*****************************************
THE FINAL VALUES

Lambda =     .346930E+003
  b    =     .138075E+003
Beta   =     .100000E+001
*****************************************
```

Table 1.4G_3 Three- and Two-Parameter Estimates for Grouping G_3

```
Input file is LNGDSET3.DAT
This is a  3 parameter fit
******************************************
THE FINAL VALUES

Lambda =    .331387E+003
  b    =    .102870E+003
Beta   =    .661333E+000
******************************************

Input file is LNGDSET3.DAT
This is a  2 parameter fit
******************************************
THE FINAL VALUES

Lambda =    .346724E+003
  b    =    .134135E+003
Beta   =    .100000E+001
******************************************
```

Table 1.4G_4 Three- and Two-Parameter Estimates for Grouping G_4

```
Input file is LNGDSET4.DAT
This is a  3 parameter fit
******************************************
THE FINAL VALUES

Lambda =    .339635E+003
  b    =    .116049E+003
Beta   =    .762044E+000
******************************************

Input file is LNGDSET4.DAT
This is a  2 parameter fit
******************************************
THE FINAL VALUES

Lambda =    .349790E+003
  b    =    .137783E+003
Beta   =    .100000E+001
******************************************
```

Table 1.4G_5 Three- and Two-Parameter Estimates for Grouping G_5

```
Input file is LNGDSET5.DAT
This is a  3 parameter fit
****************************************
THE FINAL VALUES

Lambda =     .336333E+003
  b    =     .118707E+003
Beta   =     .805587E+000
****************************************

Input file is LNGDSET5.DAT
This is a  2 parameter fit
****************************************
THE FINAL VALUES

Lambda =     .344157E+003
  b    =     .135811E+003
Beta   =     .100000E+001
****************************************
```

Table 2.1 Annual Maxima of Rainfall (in Inch) in 24 Hours at Camden Square, London, 1860-1948

GUMBEL DISTRIBUTION CALCULATIONS
Data file is GLDSET2.DAT

		3 Para Estimate		2 Para Estimate	
v	f(abs)	f(cal)	X**2	f(cal)	X**2
1	4	3.58	.05	3.86	.00
2	18	19.31	.09	18.34	.01
3	24	25.08	.05	25.07	.05
4	24	18.43	1.68	19.21	1.19
5	10	10.79	.06	11.26	.14
6	4	5.77	.54	5.82	.57
7	2	2.97	.32	2.84	.25
8	0	1.51	1.51	1.35	1.35
9	0	.76	.76	.63	.63
10	2	.38	6.79	.30	9.82
11	0	.19	.19	.14	.14
12	1	.10	8.37	.06	13.69
Totals	89	88.87		88.89	

Table 2.2 Three- and Two-Parameter Estimates

```
Input file is gldset2.dat
This is a  3 parameter fit
****************************************
THE FINAL VALUES

Lambda =     .107845E+001
  b    =     .272609E+000
Beta   =     .749054E+000
****************************************

Input file is gldset2.dat
This is a  2 parameter fit
****************************************
THE FINAL VALUES

Lambda =     .110330E+001
  b    =     .325831E+000
Beta   =     .100000E+001
****************************************
```

Table 3.1 Greatest Ages of Men

GUMBEL DISTRIBUTION CALCULATIONS
Data file is GLDSET1.DAT

		3 Para Estimate		2 Para Estimate	
v	f(abs)	f(cal)	X**2	f(cal)	X**2
1	1	1.56	.20	1.37	.10
2	8	5.57	1.06	5.94	.71
3	9	9.80	.06	10.38	.18
4	9	10.93	.34	10.93	.34
5	10	9.12	.09	8.69	.20
6	5	6.31	.27	5.90	.14
7	2	3.88	.91	3.67	.76
8	7	2.21	10.41	2.18	10.70
9	1	1.20	.03	1.25	.05
Totals	52	50.57		50.31	

Table 3.2 Three- and Two-Parameter Estimates

```
Input file is GLDSET1.DAT
This is a  3 parameter fit
*****************************************
THE FINAL VALUES

Lambda =     .988081E+002
  b    =     .224240E+001
Beta   =     .156860E+001
*****************************************

Input file is GLDSET1.DAT
This is a  2 parameter fit
*****************************************
THE FINAL VALUES

Lambda =     .986214E+002
  b    =     .171052E+001
Beta   =     .100000E+001
*****************************************
```

Table 4.1 Greatest Ensemble Ages

GUMBEL DISTRIBUTION CALCULATIONS
Data file is GLDSET3.DAT

		3 Para Estimate		2 Para Estimate	
v	f(abs)	f(cal)	X**2	f(cal)	X**2
1	1	1.99	.49	1.20	.03
2	11	7.39	1.77	8.12	1.02
3	15	15.56	.02	18.31	.60
4	18	21.27	.50	22.29	.83
5	22	20.93	.05	19.23	.40
6	15	16.04	.07	13.69	.12
7	10	10.16	.00	8.76	.18
8	8	5.57	1.06	5.28	1.41
9	3	2.73	.03	3.07	.00
10	0	1.23	1.23	1.76	1.76
11	1	.52	.44	.99	.00
Totals	104	103.39		102.70	

Table 4.2 Three- and Two-Parameter Estimates

```
Input file is GLDSET3.DAT
This is a  3 parameter fit
*****************************************
THE FINAL VALUES

Lambda =     .393298E+001
  b    =     .366330E+001
Beta   =     .389615E+001
*****************************************

Input file is GLDSET3.DAT
This is a  2 parameter fit
*****************************************
THE FINAL VALUES

Lambda =     .347478E+001
  b    =     .171638E+001
Beta   =     .100000E+001
*****************************************
```

X. PROGRAM LISTING

```
C        Research Directorate, RDEC, USA MICOM
C        Dr. C E Hall, Jr.
C        1 Sept 1988
C        mod 4 Jan 89

         IMPLICIT DOUBLE PRECISION (A-H,O-Z)
         COMMON /GGG/ NCL,FREL(225),RHO(225),XC(225),SUMA,SUMB,XBAR
         COMMON /SSS/ DSHIFT, NTOT, ALPHA, J2OR3

         CHARACTER*30 FILENAME
         CHARACTER*75 COMMENT

C        Send standard output to the printer
         OPEN(7, FILE = 'PRN')

C        Open input file
         CALL HISTORD(FILENAME,DELTAA)
C        Output the results both to the screen and to the output file
         WRITE(7,*) ' Input file is ', FILENAME
C        I/O for a two or three parameter fit
         WRITE(*,'(A\)') ' Enter desired parameter fit, 2 or 3-'
         READ(*,*) J2OR3
         WRITE(7,'(A,I2,A)') '  This is a ',J2OR3, ' parameter fit'
C        Calculate XBAR
         CALL CALXBAR

         IF (J2OR3.EQ.3) THEN
C           Calculate the roots of g(.) and h(.)
            IERR = 0
            CALL ROOTG(XROOT,YROOT,IERR)
            IF (IERR.NE.0) GOTO 900

C           Calculate the value of b
            TEMP1 = DLOG(YROOT) - PSI(YROOT)
            TEMP2 = XBAR - XROOT
            BVAL = TEMP2 / TEMP1
            RLAM = XROOT - DSHIFT
         ELSE
C           Scale BVAL to match ROOTH routine
            IF (XC(NCL).GT.70.D0) THEN
               ALPHA = XC(NCL) / 70.D0
            ELSE
               ALPHA = 1.D0
            ENDIF

C           Solve for BVAL, via ROOTH
            CALL ROOTH(X,BVAL,IERR)
            IF (IERR.NE.0) GOTO 900

C           Calculate RLAM and set BETA (YROOT)
            BVAL = BVAL * ALPHA
            SUM1 = 0.D0
            DO 100 ICNT = 1, NCL
               SUM1 = SUM1 + FREL(ICNT) * DEXP(XC(ICNT)/BVAL)
 100        CONTINUE
```

```
         ALAM = (-1.D0) * BVAL * DLOG(SUM1) - DSHIFT
         YROOT = 1.D0
      ENDIF

C     Write values to the screen
      WRITE(7,*) ' ****************************************'
      WRITE(7,*) ' THE FINAL VALUES '
      WRITE(7,*)
      WRITE(7,'( ''  Lambda = '', E15.6E3)') ALAM
      WRITE(7,'( ''    b    = '', E15.6E3)') BVAL
      WRITE(7,'( ''  Beta   = '', E15.6E3)') YROOT
      WRITE(7,*) ' ****************************************'


      GOTO 999

C     ERROR
900   CALL ERRMESS(IERR)

999   CLOSE(7)
      END

C*****************************************************************
C     This subroutine reads histogram data and calculates relative freq
      SUBROUTINE HISTORD(FILENAME,DELTAA)


      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /GGG/ NCL,FREL(225),RHO(225),XC(225),SUMA,SUMB,XBAR
      COMMON /SSS/ DSHIFT, NTOT, ALPHA, J20R3

      DIMENSION IFAB(120)

      CHARACTER*30 FILENAME

      WRITE(*,'(A\)') ' Enter the input file name -'
      READ(*,'(A)') FILENAME
      OPEN(8, FILE = FILENAME, STATUS = 'OLD')

C     Read absolute frequencies
      READ(8,'(A)') COMMENT
      READ(8,*) A, DELTAA
      READ(8,*) NUMCLASS
      READ(8,*) (IFAB(ICNT),ICNT=1,NUMCLASS)
      NCL = NUMCLASS

      NTOT = 0
      DO 100 JCNT = 1, NUMCLASS
             NTOT = NTOT + IFAB(JCNT)
100   CONTINUE

      DO 200 JCNT = 1, NUMCLASS
             FREL(JCNT) = DBLE(IFAB(JCNT)) / DBLE(NTOT)
200   CONTINUE
```

```
C       Calculate the XC
        DSHIFT = ((- DELTAA)/ 2.000)   - A
        DO 300 JCNT=1,NUMCLASS
                XC(JCNT) = DBLE(JCNT-1) * DELTAA
300     CONTINUE

        RETURN
        END

C       *****************************************************************
C       *       ERRMESS writes error messeges to the terminal
C       *****************************************************************

        SUBROUTINE ERRMESS(IERR)

        IF (IERR.EQ.0) GOTO 500

        WRITE(*,*) ' TERMINAL ERROR!!!'

        IF (IERR.EQ.1) THEN
                WRITE(*,*)' No zerocrossing found for W0'
        ELSEIF (IERR.EQ.2) THEN
                WRITE(*,*)' Root of h not bracketted'
        ELSEIF (IERR.EQ.3) THEN
                WRITE(*,*)' h does not converge to a zero'
        ELSEIF (IERR.EQ.4) THEN
                WRITE(*,*)' g-h root intersection not found'
        ENDIF

500     RETURN
        END

C       *****************************************************************
C       *       GFUNC(X,BETA) Calculates the derivative g(c,beta)
C       *       1 Sept 88
C       *       mod 2 Dec 88
C       *       Res. Dir., C. E. Hall, Jr.
C       *****************************************************************

        FUNCTION GFUNC(X,BETA)

        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        COMMON /GGG/ NCL,FREL(225),RHO(225),XC(225),SUMA,SUMB,XBAR
        COMMON /SSS/ DSHIFT, NTOT, ALPHA, J20R3

        XSHIFT = X
        CALL SUMAB(XSHIFT,BETA)
        SBETA = DLOG(BETA) - PSI(BETA)

        GFUNC = (DBLE(NTOT)*XBAR-SUMB)*BETA*SBETA -
     -  (DBLE(NTOT)*XBAR - DBLE(NTOT)*XSHIFT)

        RETURN
        END
```

```
C         *****************************************************************
C         *        HFUNC(BETA) Calculates the derivative h(c,beta)
C         *        1 Sept 88
C         *        13 Oct 88 mod for three function root finding
C         *        Res. Dir., C. E. Hall, Jr.
C         *****************************************************************

          FUNCTION HFUNC(X,BETA)

          IMPLICIT DOUBLE PRECISION (A-H,O-Z)
          COMMON /GGG/ NCL,FREL(225),RHO(225),XC(225),SUMA,SUMB,XBAR
          COMMON /SSS/ DSHIFT, NTOT, ALPHA, J2OR3

          XSHIFT = X

          IF (J2OR3.EQ.3) THEN
                  SBETA = DLOG(BETA) - PSI(BETA)
                  YLOG =  SBETA /  (XSHIFT - XBAR)

                  SUM = 0.D0
                  DO 100 JCNT=1,NCL
                    SUM = SUM+FREL(JCNT)*DEXP(YLOG*(XC(JCNT)-XC(1)))
100               CONTINUE

                  HFUNC = DEXP(YLOG*(XSHIFT-XC(1))) - SUM

          ELSEIF (J2OR3.EQ.2) THEN
                  BVAL = BETA * ALPHA
                  TEMP1 = 0.D0
                  TEMP2 = 0.D0

                  DO 200 JCNT = 1,NCL
                          TERM = FREL(JCNT) / DEXP(XC(JCNT)/BVAL)
                          TEMP1 = TEMP1 + TERM
                          TEMP2 = TEMP2 + TERM * XC(JCNT)
200               CONTINUE

                  HFUNC = XBAR - BVAL - TEMP2 / TEMP1
          ENDIF

          RETURN
          END

C         *****************************************************************
C         *        SUMAB(X,BETA) Calculates the series sums A and B
C         *        1 Sept 88
C         *        Res. Dir., C. E. Hall, Jr.
C         *****************************************************************

          SUBROUTINE SUMAB(X,BETA)

          IMPLICIT DOUBLE PRECISION (A-H,O-Z)
```

```
      COMMON /GGG/ NCL,FREL(225),RHO(225),XC(225),SUMA,SUMB,XBAR
      COMMON /SSS/ DSHIFT, NTOT, ALPHA, J2OR3

      XSHIFT = X
      SBETA = DLOG(BETA) - PSI(BETA)
      SUMA = 0.D0
      SUMB = 0.D0

      DO 410 J = 1,NCL

              SUMA = SUMA + DBLE(NTOT)*FREL(J) * DEXP( ( (XSHIFT - XC(J) )
     -               (XBAR - XSHIFT) ) * SBETA )

              SUMB = SUMB + DBLE(NTOT)*FREL(J) * XC(J) * DEXP(
     -              ( (XSHIFT - XC(J)) / (XBAR - XSHIFT) ) * SBETA )

410   CONTINUE

      RETURN
      END

C     ******************************************************************
C     PSI2 CALCULATES VALUE OF PSI FUNTION FOR GIVEN ARGUMENT
C     ******************************************************************
      FUNCTION PSI(YTX)
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      DATA U,T1,T2,T3,T4,T5/1D0,12D0,10D0,21D0,20D0,11D0/
      A=0.D0

      XTX = YTX

      goto 200

100   A=A+U/XTX
      XTX=XTX+U
200   IF (XTX.LE.2D1) goto 100

      Q=U/(XTX*XTX)
      PSI=Q*(Q*(Q*(-Q/T5+U/T4)-U/T3)+U/T2)
      PSI=DLOG(XTX)-A-U/(2D0*XTX)+(PSI-U)*Q/T1
      RETURN
      END

*************************************************************************

      SUBROUTINE SETLIM(XMIN,XCEN,XMAX,DELXPOS,DELXNEG)

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      COMMON /SSS/ DSHIFT, NTOT, ALPHA, J2OR3
      COMMON /GGG/ NCL,FREL(225),RHO(225),XC(225),SUMA,SUMB,XBAR

      XMIN = 0.D0
      XMAX = XBAR
      XCEN = XMAX - 0.5D0 * ( XC(2)-XC(1) )
```

```
      DELXPOS = XMAX - XCEN
      DELXNEG = XCEN - XMIN

      RETURN
      END

C     ***************************************************************
C     *      CALXBAR(.) calculates XBAR for the gumbel
C     *      distribution
C     *      2 Dec 88
C     *      C E Hall, Jr, Research Directorate, HDEC, MICOM
C     ***************************************************************

      SUBROUTINE CALXBAR

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /GGG/ NCL,FREL(225),RHO(225),XC(225),SUMA,SUMB,XBAR
      COMMON /SSS/ DSHIFT, NTOT, ALPHA, J20R3

C     Calculate the moments
      XBAR = 0.D0
      DO 100 ICNT=1,NCL
             XBAR = XBAR + FREL(ICNT) * XC(ICNT)
100   CONTINUE

      RETURN
      END
      C      ***************************************************************
C     *      ROOTG(XROOT,YROOT,IERR)
C     *      31 Aug 88
C     *      Res. Dir., C E Hall, Jr.
C     ***************************************************************

      SUBROUTINE ROOTG(XROOT,YROOT,IERR)

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      CALL RTGBRACK(XA,YA,XB,YB,IERR,I0FLAG)
      IF (I0FLAG.NE.0) GOTO 100
      IF (IERR.NE.0 ) GOTO 200

      CALL BIS4RTG(XA,YA,XB,YB,XROOT,YROOT,IERR)

      GOTO 200

100   XROOT = XA
      YROOT = YA

200   RETURN
      END

C     ***************************************************************
C     *      CVALUE(JCNT,DELXPOS,DELXNEG,XPOS,XNEG)
C     *      This subroutine calculates the positive and negative
C     *      shift values.
```

```
C        *       31 Aug 88
C        *       Res. Dir., C. E. Hall, Jr.
C        *************************************************************

         SUBROUTINE CVALUE(JCNT,DELXPOS,DELXNEG,XPOS,XNEG,XCEN)

         IMPLICIT DOUBLE PRECISION (A-H,O-Z)

         TERM = DEXP( DBLE(JCNT) *DLOG(2.0D0) )

         XPOS = XCEN + TERM * DELXPOS / ( TERM + 99.0D0 )
         XNEG  = XCEN - TERM * DELXNEG  / ( TERM + 99.0D0 )

         RETURN
         END

C        *************************************************************
C        *       RTGBRACK(.)
C        *       This subroutine brackets the root of the function g
C        *       31 Aug 88
C        *       Res. Dir., C. E. Hall, Jr.
C        *************************************************************

         SUBROUTINE RTGBRACK(XA,YA,XB,YB,IERR,I0FLAG)

         IMPLICIT DOUBLE PRECISION (A-H,O-Z)

         JCNT = 0
         I0FLAG = 0
         IRLFLG = 0
         MAXJ = 30
         CALL SETLIM(XMIN,XCEN,XMAX,DELXPOS,DELXNEG)
         XC = XCEN

         CALL ROOTH(XC,YC,IERR)
         IF(IERR.NE.0) GOTO 400
         ZC = GFUNC(XC,YC)
         IF (ZC.EQ.0.D0) GOTO 390

C        Calculation loop
200      JCNT = JCNT + 1
         CALL CVALUE(JCNT,DELXPOS,DELXNEG,XPOS,XNEG,XCEN)

C        Check positive c shift
         IF ((IRLFLG.EQ.2).OR.(IRLFLG.EQ.3)) GOTO 225
         CALL ROOTH(XPOS,YP,IERR)
         IF (IERR.NE.0) THEN
                 IERR = 0
                 IRLFLG = IRLFLG + 2
                 GOTO 225
         ENDIF

         ZP = GFUNC(XPOS,YP)
         IF (ZP.EQ.0.D0) GOTO 390
         IF ((ZP*ZC).LT.0.D0) GOTO 350
```

```
C       Check negative c shift
225     IF ((IALFLG.EQ.1).OR.(IALFLG.EQ.3)) GOTO 250
        CALL ROOTH(XNEG,YN,IERR)
        IF (IERR.NE.0) THEN
                IERR = 0
                IALFLG = IALFLG + 1
                GOTO 250
        ENDIF

        ZN = GFUNC(XNEG,YN)
        IF (ZN.EQ.0.D0) GOTO 395
        IF ((ZN*ZC).LT.0.D0) GOTO 360

250     IF ((IALFLG.NE.3).AND.(JCNT.LT.MAXJ)) GOTO 200

        IERR = 4
        GOTO 400

C       Sign change in positive j-1,j bracket
350     XB = XPOS
        YB = YP
        JMIN1 = JCNT -1
        CALL CVALUE(JMIN1,DELXPOS,DELXNEG,XA,XNEG,XCEN)
        CALL ROOTH(XA,YA,IERR)
        GOTO 400

C       Sign change in negative j,j-1 bracket
360     XA = XNEG
        YA = YN
        JMIN1 = JCNT - 1
        CALL CVALUE(JMIN1,DELXPOS,DELXNEG,XPOS,XB,XCEN)
        CALL ROOTH(XB,YB,IERR)
        GOTO 400

C       Zero found
390     I0FLAG = 1
        XA = XPOS
        YA = YP
        GOTO 400

C       Zero found
395     I0FLAG = 1
        XA = XNEG
        YA = YN

400     RETURN
        END

C       ****************************************************************
C       *       BIS4RTG - Bisects the interval until delta X, and
C       *               delta Y are both less than 1.0E-7
C       *       31 Aug 88
C       *       Res. Dir., C. E. Hall, Jr.
C       ****************************************************************
```

```
      SUBROUTINE BIS4RTG(XA,YA,XB,YB,XM,YM,IERR)

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      TOL = 1.0D-07

      ZA = GFUNC(XA,YA)
      ZB = GFUNC(XB,YB)

C     Bisection loop
100   XC = (XA + XB) / 2.0D0
      CALL ROOTH(XC,YC,IERR)
      IF (IERR.NE.0) GOTO 300

      ZC = GFUNC(XC,YC)
      IF (ZC.EQ.0.D0) GOTO 250

      IF ((ZC*ZA).GT.0.D0) THEN
              XA = XC
              YA = YC
              ZA = ZC
      ELSE
              XB = XC
              YB = YC
              ZB = ZC
      ENDIF

      DELTAX = DABS(XB-XA)
      DELTAY = DABS(YB-YA)

      IF( (DELTAX.GE.TOL) .OR. (DELTAY.GE.TOL) ) GOTO 100

      XM = (XA + XB) / 2.D0
      YM = (YA + YB) / 2.D0
      GOTO 300

C     Exact zero found
250   XM = XC
      YM = YC

300   RETURN
      END

C     ************************************************************
C     *       RTHBRACK - Root H Bracketing Subroutine
C     *       This subroutine brackets the root of only the H function
C     *       31 Aug 88
C     *       Res. Dir., Dr. C. E. Hall, Jr.
C     ************************************************************

      SUBROUTINE RTHBRACK(XV,YA,ZA,YB,ZB,IERR)

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

C     Set starting parameters
```

```
      Y0 = 0.1D0
      DUPFAC = 1.618D0
      MAXITER = 19
      IERR = 0

C     Set initial values
      YA = Y0
      ZA = HFUNC(XV,YA)
      ITER = 0

C     Evaluation loop
100   ITER = ITER + 1
      YB = YA * DUPFAC
      ZB = HFUNC(XV,YB)

      IF (ZB.EQ.0.D0) GOTO 150
      IF ((ZA*ZB).LT.0.D0) GOTO 150

      YA = YB
      ZA = ZB

      IF (ITER.LT.MAXITER) GOTO 100

C     MAXITER exceeded set error flag
      IERR = 4

150   RETURN
      END

C     ***************************************************************
C     *       ROOTH - Finds the root of h
C     *       31 Aug 88
C     *       Res. Dir., Dr. C. E. Hall, Jr.
C     ***************************************************************

      SUBROUTINE ROOTH(XV,YROOT,IERR)

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      CALL RTHBRACK(XV,YA,ZA,YB,ZB,IERR)
      IF(IERR.NE.0) GOTO 200
      CALL BROOT(XV,YA,ZA,YB,ZB,YROOT,IERR)

200   RETURN
      END

*-----------------------------------------------------------------
      SUBROUTINE BROOT(XV,YA,ZA,YB,ZB,YROOT,IERR)
      IMPLICIT DOUBLE PRECISION (P-Z)

      DOUBLE PRECISION EPSI,HFUNC
      PARAMETER (MAXIT=100,EPSI=3.D-8)
      ZC=ZB
      DO 510 NIT=1,MAXIT
        IF (ZB*ZC.GT.0D0) THEN
```

```
          YC=YA
          ZC=ZA
          YD=YB-YA
          YE=YD
        ENDIF
        IF(DABS(ZC).LT.DABS(ZB)) THEN
          YA=YB
          YB=YC
          YC=YA
          ZA=ZB
          ZB=ZC
          ZC=ZA
        ENDIF
        TOL1=2.D0*EPSI*DABS(YB)+.5D0*EPSI
        YM=.5D0*(YC-YB)
        IF (DABS(YM).LE.TOL1.OR.ZB.EQ.0D0) THEN
          YROOT=YB
          RETURN
        ENDIF
        IF(DABS(YE).GE.TOL1.AND.DABS(ZA).GT.DABS(ZB)) THEN
          S=ZB/ZA
          IF(YA.EQ.YC) THEN
            T=2D0*YM*S
            Q=1D0-S
          ELSE
            Q=ZA/ZC
            R=ZB/ZC
            T=S*(2D0*YM*Q*(Q-R)-(YB-YA)*(R-1.D0))
            Q=(Q-1.D0)*(R-1.D0)*(S-1.D0)
          ENDIF
          IF(T.GT.0.D0) Q=-Q
          T=DABS(T)
          Y3=3D0*YM*Q-DABS(TOL1*Q)
          IF(2.D0*T.LT.DMIN1(Y3,DABS(YE*Q))) THEN
            YE=YD
            YD=T/Q
          ELSE
            YD=YM
            YE=YD
          ENDIF
        ELSE
          YD=YM
          YE=YD
        ENDIF
        YA=YB
        ZA=ZB
        IF(DABS(YD).GT.TOL1) THEN
          YB=YB+YD
        ELSE
          YB=YB+DSIGN(TOL1,YM)
        ENDIF
        ZB=HFUNC(XV,YB)
  510 CONTINUE
      IERR=5
      RETURN
      END
```

```
C       *******************************************************************
C       *       This program calculates the gumbel pdf from the
C       *       gumbel parameters
C       *       powell's method.
C       *       Dr. C. E. Hall, Jr.
C       *       Res. Dir., RDEC, MICOM
C       *       17 Oct 1988
C       *       mod 5 Dec 1988
C       *******************************************************************

        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        CHARACTER*30 FILENAME

        COMMON /GGG/ NCL, FREL(225), RHO(225), XC(225)
        COMMON /SSS/ XSHIFT, NTOT, J2OR3

        OPEN(7,FILE='PRN')

        WRITE(*,*) ' 3 Parameters'
        WRITE(*,'(A\)') ' Enter the Lambda-'
        READ(*,*) RLAM0

        WRITE(*,'(A\)') ' Enter b-'
        READ(*,*) SIGMA0

        WRITE(*,'(A\)') ' Enter Beta-'
        READ(*,*) RMU0


        WRITE(*,*) ' 2 Parameters'
        WRITE(*,'(A\)') ' Enter the Lambda-'
        READ(*,*) RLAM1

        WRITE(*,'(A\)') ' Enter b-'
        READ(*,*) SIGMA1

        RMU1 = 1.D0

C       Set parameter for unshift XC's
        J2OR3 = 2
        XSHIFT = 0.D0

        CALL HISTORD(FILENAME,DELTAA)

C       WRITE HEADER TO PRINTER
        WRITE(7,*) '                GUMBEL DISTRIBUTION CALCULATIONS'
        WRITE(7,'(A,1X,A)') '                      Data file is ',FILENAME
        WRITE(7,'(1X)')
        WRITE(7,900)
900     FORMAT (1X,30X,'3 Para Estimate',7X,'  2 Para Estimate')
        WRITE(7,910)
910     FORMAT (13X,1Hv,5X,6Hf(abs),5X,6Hf(cal),6X,4HX**2,6X,
     -    6Hf(cal),6X,4HX**2)
        ISY1 = 0
        SY2 = 0.D0
        SY3 = 0.D0
```

```
      DO 500 ICNT=1,NCL

              Y1 = DBLE(NTOT) * FREL(ICNT)
              Y2 = DBLE(NTOT)*PDF(XC(ICNT),ALAM0,SIGMA0,AMU0)*DELTAA
              Y3 = DBLE(NTOT)*PDF(XC(ICNT),ALAM1,SIGMA1,AMU1)*DELTAA
              SY2 = SY2 + Y2
              SY3 = SY3 + Y3
              DEL1 = (Y1 - Y2) * (Y1 - Y2) / Y2
              DEL2 = (Y1 - Y3) * (Y1 - Y3) / Y3
              IY1 = NINT(Y1)
              ISY1 = ISY1 + IY1
              WRITE(7,901) ICNT, IY1,Y2,DEL1,Y3,DEL2
901           FORMAT (12X,I3,7X,I3,1X,4(5X,F6.2))

500   CONTINUE

      WRITE (7,902) ISY1,SY2,SY3
902   FORMAT(12X,'Totals'4X,I3,6X,F6.2,16X,F6.2)
      WRITE(7,'(1H1)')
      CLOSE(7)
      END

C     ************************************************************
C     This subroutine reads histogram data and calculates relative freq
      SUBROUTINE HISTORD(FILENAME,DELTAA)


      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /GGG/ NCL,FREL(225),RHO(225),XC(225)
      COMMON /SSS/ XSHIFT, NTOT, J2OR3
      DIMENSION IFAB(120)

      CHARACTER*30 FILENAME

      WRITE(*,'(A\)') ' Enter the input file name -'
      READ(*,'(A)') FILENAME
      OPEN(8, FILE = FILENAME, STATUS = 'OLD')

C     Read absolute frequencies
      READ(8,'(A)') COMMENT
      READ(8,*) A, DELTAA
      READ(8,*) NUMCLASS
      READ(8,*) (IFAB(ICNT),ICNT=1,NUMCLASS)
      NCL = NUMCLASS

      NTOT = 0
      DO 100 JCNT = 1, NUMCLASS
              NTOT = NTOT + IFAB(JCNT)
100   CONTINUE

      DO 200 JCNT = 1, NUMCLASS
              FREL(JCNT) = DBLE(IFAB(JCNT)) / DBLE(NTOT)
200   CONTINUE
```

```
C       Calculate the XC
        IF (J2OR3.EQ.2) THEN
                XSHIFT = (-1.D0) * XSHIFT
                DO 250 ICNT = 1, NUMCLASS
                        XC(ICNT)=A+(DBLE(ICNT)-0.5D0)*DELTAA+XSHIFT
250             CONTINUE
        ELSE
                XSHIFT = (( - DELTAA)/ 2.0D0)  - A
                DO 300 JCNT=1,NUMCLASS
                        XC(JCNT) = DBLE(JCNT-1) * DELTAA
300             CONTINUE
        ENDIF

        RETURN
        END

C       ******************************************************************
C       *       The Gumbel Probability Density function
C       ******************************************************************

        FUNCTION PDF(X,RLAMDA,BB,BETA)

        IMPLICIT DOUBLE PRECISION (A-H,O-Z)

        XI = (X - RLAMDA ) / BB

        F1 = BETA * DLOG(BETA) - DLOG(BB) - GALOG(BETA)
        F2 = BETA * ( DEXP( -1.D0 * XI ) + XI )

        PDF = DEXP( F1 - F2 )

        RETURN
        END

C       ******************************************************************
C       *       LN GAMMA
C       ******************************************************************
        FUNCTION GALOG(QTEMP)

        IMPLICIT DOUBLE PRECISION (G,Q-Z)
        PARAMETER (U=1.D0,XMLIM=14.D0)
        GALOG = 999.9999

        IF(QTEMP.EQ.0.D0) RETURN
        X = QTEMP
        XMUL=U
110     XMUL = XMUL * X
        X = X + U
        IF(X.LT.XMLIM) GOTO 110
        Q = X * X
        Y=(U/21.D0+((-1.D0)/28.D0+5.D0/(99.D0*Q))/Q)/Q
        Y=(5.D0+(U/(-6.D0)+Y)/Q)/(60.D0*X)-X*(U-DLOG(X))
        GALOG=Y+DLOG(DSQRT(8.D0*DATAN(U)/X))-DLOG(XMUL)
        RETURN
        END
```

REFERENCES

1. Brooks, C. E. P. and Carruthers, N., Handbook of Statistical Methods in Meteorology, Her Majesty's Stationary Office, London, 1953.

2. Doetsch, G., Handbuch der Laplace-Transformation, Vol. I, Birkhäuser, Basel, 1950.

3. Gradshteyn, I.S. and Ryzhik, I. M., Table of Integrals, Series, and Products, Academic Press, New York, 1965.

4. Gumbel, E. J., Les plus grands âges en Suisse, J. Statist. Rev. Econ. Suisse 70 Fasc. 4 (1934), 602-614.

5. Hahn, G. H. and Shapiro, S. S., Statistical Models in Engineering, John Wiley, New York, 1967.

6. Johnson, N. L. and Kotz, S., Continuous Univariate Distributions, Vol. 1, Houghton Mifflin, Boston, 1970.

7. Lambert, J. A., "Estimation of Parameters in the Three-Parameter Lognormal Distribution," Australian J. of Stat. 6 (1964), 29-32.

8. Law, A. M. and Kelton, W. D., Simulation Modeling and Analysis, McGraw-Hill, New York, 1982.

9. Lehnigk, S. H., "Initial Condition Solutions of the Generalized Feller Equation," J. Appl. Math. Phys. (ZAMP) 29 (1978), 273-294.

10. Lehnigk, S. H., "On a Class of Probability Distributions," Math. Meth. in the Appl. Sci. 9 (1987), 210-219.

11. Lehnigk, S. H., "Maximum-Likelihood Estimation of the Parameters of a Four-Parameter Class of Probability Distributions," Proc. Edin. Math. Soc. 31 (1988), 271-283.

12. Lehnigk, S. H., "Conservative Similarity Solutions of the One-Dimensional Autonomous Parabolic Equation," J. Appl. Math. Phys. (ZAMP) 27 (1976), 385-391.

13. Press, V. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T., Numerical Recipes, Cambridge University Press, New York, 1987.

14. Roussas, G. G., A First Course in Mathematical Statistics, Addison-Wesley, Reading, 1973.

15. Whittaker, E. T. and Watson, G. N., A Course of Modern Analysis, Cambridge University Press, Cambridge, 1958.

DISTRIBUTION

	Copies
Mr. Carl B. Bates Math Stat Team, CSCA-ASA US Army Concepts Analysis Agency 8120 Woodmont Avenue Bethesda, MD 20814-2797	1
COL Frank Giordano Department of Mathematics US Military Academy West Point, NY 10996	1
Dr. Dennis M. Tracey Mechanics of Materials Division US Army Materials & Mechanics Research Center Watertown, MA 02172	1
Director Ballistic Research Laboratory ATTN: AMXBR-SECAD, Dr. Malcolm Taylor Aberdeen Proving Ground, MD 21005-5066	1
Commander Harry Diamond Laboratories ATTN: DELHD-NW-RA (Dr. Timothy M. Geipe) 2800 Powder Mill Road Adelphi, MD 20783-1197	1
Commander Materiel Readiness Support Activity ATTN: AMXMD-E Mr. Willard F. Stratton Chief, Readiness Division Lexington, Kentucky 40511-5101	1
Commander US Army Tank-Automotive Command ATTN: AMSTA-ZSS (Dr. William Jackson) Warren, MI 48397-5000	1
Commander Instrumentation Directorate ATTN: STEWS-ID-P (Mr. Robert E. Green) White Sands Missile Range, NM 88002	1
Commander US Army Materiel Systems Analysis Activity ATTN: AMXSY-MP (Mr. Herbert Cohen) Aberdeen Proving Ground, MD 21005-6577	1

DISTRIBUTION (Continued)

	Copies
Dr. Douglas B. Tank Chief, Department of Biostatistics/Applied Math Division of Biometrics & Medical Info Processing Walter Reed Army Institute of Research Washington, DC 20012	1
Dr. Edmund H. Inselmann US Army Combined Arms Studies & Analysis Activity ATTN: ATOR-CAQ Ft. Leavenworth, KS 66027-5230	1
Dr. Carl Russell ATTN: CSTE-ASO-A USA Operational Test & Evaluation Agency 5600 Columbia Pike Falls Church, VA 22041-5115	1
IIT Research Institute ATTN: GACIAC 10 W. 35th Street Chicago, IL 60616	1
US Army Materiel System Analysis Activity ATTN: AMXSY-MP Aberdeen Proving Ground, MD 21005	1
Dr. Guttalu R. Viswanath Department of Mathematics and Computer Science South Carolina State College Orangeburg, SC 29117	10
AMSMI-RD, Dr. McCorkle Dr. Stephens Dr. Rhoades	1
AMSMI-RD-SS, Dr. Grider	2
AMSMI-RD-ST, Dr. Mixon	2
AMSMI-RD-TE	2
AMSMI-RD-BA, Mr. Ciliax	2
AMSMI-RD-RE-OP, Dr. Lehnigk	10
AMSMI-RD-RE-OP, Dr. Hall	10
AMSMI-RD-CS-R	15
AMSMI-RD-CS-T	1
AMSMI-GC-IP, Mr. Bush	1